Zadanie č. 3

Strojové učenie a neurónové siete

Bc. Miroslav Malíšek

# Analýza zadania

Ibbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbeddddddddddddddddddddddddddddddddddddddddddddddfvhrficvorocnoonkocnrkcnrkcnrkcnrkcnrkcnrcknr

# Časť zadania (Spoznajte dáta)

Obrázky, ktoré sa použijú v tomto zadaní, sú rozdelené do dvoch množín (priečinkov) – trénovacie a testovacie. V oboch množinách sa nachádzajú rovnaké triedy, resp. typy zvierat. Celkovo ich je 90. Ku každému zo zvierat máme celkovo k dispozícii 60 obrázkov, z toho 54 je trénovacích a 6 testovacích. Spolu je to 5400 obrázkov. Tieto informácie som v kóde uložil do dataframe-u, aby to bolo prehľadné.

Nasledujúca koláž zobrazuje reprezentanta každej triedy. Zvolil som ho ako prvý obrázok každej triedy trénovacej množiny.

Obrázok, na ktorom je koláž, zbierka, fotomontáž, snímka obrazovky

Automaticky generovaný popis

## *Analýza podľa ImageNet*

Ako model som si vybral EfficientNetB2, ktorá dosahuje úspešnosť 80.1%, resp. 94.9% v TOP-5. Zároveň obsahuje 9.2 miliónov parametrov, čo je rozumné číslo vzhľadom na dostupný hardvér (tento model použijeme aj v ďalších častiach zadania).

Predikcie získam zo všetkých dostupných obrázkov (trénovacie aj testovacie), takže vyhodnotenie prebieha na 5400 obrázkoch.

Nakoľko model nie je prispôsobený na naše výstupy, nemohol som priamo porovnávať predikované zaradenie obrázku do triedy s reálny zaradením. Často sa totiž stalo, že model to vyhodnotil správne, akurát názov zvieraťa bol iný, ako u nás. Prípadne predikoval už aj poddruh zvieraťa. Ak by sme to priamo porovnávali, prišli by sme o cenné informácie, ktoré vieme získať len ručným vyhodnotením.

Vyhodnotil som to teda tak, že pre každú našu triedu som zobrazil 3 najpočetnejšie predikcie modelu (resp. menej ak model predikoval pre našu triedu všetko do jednej, resp. dvoch tried). Výsledky som uložil do dataframe-u, ktorý vyzerá nasledovne:

Obrázok, na ktorom je text, snímka obrazovky, jedálny lístok, softvér

Automaticky generovaný popisObrázok, na ktorom je text, snímka obrazovky, jedálny lístok

Automaticky generovaný popisObrázok, na ktorom je text, jedálny lístok, snímka obrazovky

Automaticky generovaný popis

Obrázok, na ktorom je text, snímka obrazovky, jedálny lístok

Automaticky generovaný popisObrázok, na ktorom je text, snímka obrazovky, jedálny lístok

Automaticky generovaný popisObrázok, na ktorom je text, snímka obrazovky, jedálny lístok, softvér

Automaticky generovaný popisObrázok, na ktorom je text, snímka obrazovky, jedálny lístok, softvér

Automaticky generovaný popisObrázok, na ktorom je text, snímka obrazovky, jedálny lístok, softvér

Automaticky generovaný popis

Stĺpec vľavo reprezentuje naše triedy zvierat (pravdivé), stredný predpovedané modelom a stĺpec vpravo zobrazuje, koľkokrát model predpovedal hodnotu v strednom stĺpci pre pravdivú triedu zvierat vľavo.

Pre každé zviera sme mohli mať maximálny počet predpovedí 60, takže najlepšie je, ak je číslo vpravo práve 60. To sa podarilo v jedinom prípade a to pre zviera *wombat*. Iné predikcie teda pre toto zviera už model nedal. Pre dve zvieratá predpovedal model 59x rovnakú triedu, konkrétne pre *orangutan* a*panda*. V prípade pandy však vidíme, že model predpovedal „giant panda“, takže tento prípad potvrdzuje to, že sme to nemohli presne porovnávať. Vyšlo by nám, že pre pandu sa model netrafil ani raz, čo by nebola pravda. V rozmedzí 50 až 58 model predpovedal rovnakú triedu v týchto zvieratách: *badger, bison, boar, chimpanzee, cockroach, coyote, eagle, flamingo, fly, goldfish, goose, gorilla, hare, hippopotamus, hornbill, hummingbird, hyena, jellyfish, koala, leopard, lion, lobster, otter, pelican, penguin, pig, porcupine, squirrel, starfish, tiger, zebra.* Spolu teda model predpovedal 50 alebo viackrát rovnako pri 34 zvieratách.

Pri hlbšej analýze by sme ešte zistili, že sú zvieratá, pri ktorých model síce nepredpovedal až tak veľakrát jednu triedu, avšak predpovedal už aj druhy jednotlivých zvierat, ako napríklad v prípade *bear, crab, elephant* alebo *shark*. Ak by sme brali druhy ako jedno zviera, tak v týchto prípadoch by sme sa tiež dostali nad 50 správnych predpovedí.

Na druhú stranu, pri niektorých zvieratách sa model výrazne mýlil. Uvediem len tie prípady, v ktorých sa v troch najčastejších predpovediach nevyskytuje ani raz správna predpoveď. Sú to *bat cow, donkey, goat, mosquito, mouse, octopus, okapi, oyster, pigeon, possum, raccoon, rat, rhinoceros, seahorse, squid, woodpecker.* Je pravdepodobné, že model žiadne obrázky podobné týmto zvieratám predtým nevidel a teda nebol ich schopný správne zaradiť.

Medzi najviac podobné triedy podľa predpovedí patria *antelope – deer* (predpovede gazelle), *starfish – octoupus* (predpovede starfish), *badger – possum* (predpovede badger), *dolphin – whale* (predpovede gray whale a killer whale), *cow – ox* (predpovede ox). V niektorých prípadoch sú podobnosti zjavné (*antelope – deer, badger – possum*). V iných prípadoch je to ale najmä preto, lebo jedna z tých tried nie je na modeli natrénovaná a nemá svoju vlastnú triedu. Model ju teda priradil do inej triedy na základe nejakých znakov (*starfish – octopus* sú vodné zvieratá, *cow* nemá vlastnú triedu a najviac sa podobá z modelu práve na *ox*).

# Časť zadania

V tejto časti vytvárame a trénujeme konvolučnú neurónovú sieť na klasifikáciu zvierat.

Na vytvorenie trénovacej, validačnej aj testovacej množiny použijeme funkciu image\_dataset\_from\_directory() z knižnice Keras. Validačné dáta vytvoríme z trénovacej množiny v pomere 1:9. Veľkosti obrázkov nastavíme na 180x180 pixelov a batch size na 32. Trénovacie dáta zamiešame. Ako samotné obrázky použijeme tie dodané so zadaním, pričom vo vyššie spomenutej funkcii sa využíva to, že sú uložené do priečinkov.

Nasledujúci kód teda slúži na vytvorenie generátoru dát:

Obrázok, na ktorom je text, snímka obrazovky, písmo

Automaticky generovaný popis

Následne si vytvoríme konvolučnú neurónovú sieť. Bude obsahovať 2 konvolučné vrstvy, čo znamená 2 vrstvy Conv2D + MaxPooling2D.

V prvej vrstve som zvolil počet filtrov 16 a kernelovú dimenziu 3x3. V druhej vrstve zase 32 filtrov, kernelová dimenzia ostala rovnaká. Aktivačná funkcia bude v oboch vrstvách relu. Maximálnu veľkosť poolingu som v oboch vrstvách zvolil 3x3.

Nasleduje vrstva Flatten, ktorá predchádza plne prepojeným vrstvám.

Plne prepojené vrstvy sú tvorené vrstvami typu Dense, pričom ich počet a aj počet neurónov v každej z nich už bude závisieť od konkrétnej realizácie siete. Budú tú uvedené rôzne konfigurácie. Avšak každá skrytá vrstva má aktivačnú funkciu relu a posledná, výstupná má aktivačnú funkciu softmax s počtom neurónov 90 (toľko máme rôznych druhov zvierat, resp. tried).

############################################

Kriteriálna funkcia je použitá sparse\_categorical\_crossentropy. Je to kvôli tomu, že sa jedná o viactriednu klasifikáciu. Od obyčajnej categorical\_crossentropy

###########################################

Dropout(0.25), 64, 128, 128, 64 – iba trošku pretrénované